REMARKS

This amendment is in response to the Examiner's Office Action dated 12/8/2003. Applicant is appreciative for the recognized allowable subject matter. This amendment should obviate outstanding issues. Reconsideration of this application is respectfully requested in view of the foregoing amendment and the remarks that follow.

STATUS OF CLAIMS

Claims 1-46 are pending.

Claims 4 and 12-13 stand rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claims 1, 3-4, 10-13, 18-22, 24-25, 31-34, and 39-45 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Fernandez et al. (USP 6,604,100).

Claims 2, 5-9, 14-17, 23, 26-30, 35-38, and 46 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

OVERVIEW OF CLAIMED INVENTION

The presently claimed invention provides a method of translating queries of XML data into queries against a relational database. In addition, a method is provided for publishing the queried relational data as an XML document thus precluding the necessity for an intermediate data representation.

Translation occurs when an XML query is parsed and transformed into an intermediate language-neutral query representation. The intermediate language-neutral query representation is then transformed into an SQL query that is issued over a relational table in the database where the relational table corresponds to a virtual XML document.

A tagger graph is generated from the intermediate language-neutral representation of the XML query. Each of the nodes of the tagger graph is an operator that performs processing on the results of the SQL query results. Traversing the tagger graph generates tags and structure for XML output. When SQL query results from the relational database are provided as input to the tagger graph and a traversal of the graph generates the tags and the structure by processing these inputs, a structured XML document is produced.

Based on the structure of the initial XML query, a tagger graph is generated during run-time to put currently flattened relational tuples back in a hierarchical graphical XML structure.

REJECTIONS UNDER 35 U.S.C. § 112, SECOND PARAGRAPH

The examiner has rejected claims 4 and 12-13 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Specifically, claims 4 and 12-13 have been rejected for lacking proper antecedent basis in "said tagger graph". Applicants have amended dependent claims 4, 12, and 13 to provide proper antecedent basis. It is, therefore, respectfully requested that the §112 rejection be removed.

REJECTIONS UNDER 35 U.S.C. § 103(a)

The Fernandez et al reference (hereafter Fernandez) provides for a method to convert an XML-QL user query over an XML view to a Relational to XML transformation language (RXL)

query, converts an RXL query to an SQL query, and retrieves XML data stored in a relational database. Fernandez also proposes an XML generator for converting flat results (streams of tuples) into XML. The primary basis for this reference is its provision for the storage of XML data in view of a relational database - no major mention of how an XML generator formats and renders resultant XML data is made.

With regards to claims 1, 22, 44, and 45, the examiner has included the Fernandez reference specifically for its discussion of retrieval of XML data stored in relational databases and subsequent conversion of relational data to XML. The examiner has pointed to col. 3, lines 10-24 as constituting a tagger tree graph having each node comprising a tagger operator with a parse tree associated, therewith. However, a closer reading of this section reveals only a general overview of XML query formulation for the purpose of XML data extraction, which is well known in the art.

While the disclosed method (Fernandez) provides for an executable query produced by a query composer module composition of an RXL view query and an XML-QL user query, as cited by the examiner in col. 4, line 64 through col. 5, line 25, the result of an executable query typically includes only a small fragment of the database, thus producing an XML document. This is clearly different from the method of the present invention, wherein a tagger tree graph is generated to tag flat, relational database tuples resulting from an XML query. The present invention discloses a default XML view upon which more complex views are derived – the tagger tree graph corresponding to a particular XML view is derived at run-time from an XML query directly. In this manner, any hierarchical structure specified by an initial XML query is supported. The method disclosed by Fernandez does not mention how relational data is converted to XML, nor does it mention the module facilitating its conversion is generated at run-time.

Additionally, the examiner has also pointed to col. 12, lines 32-45 wherein Fernandez discloses the determination of a composition query as a result of evaluating user queries on view query templates. Fernandez further discloses a solutions relation tuple representing a matching user query pattern and view query template pair. Fernandez does not teach nor suggest a system for tagging or structuring the results of an SQL query of XML data stored in a relational database. In fact, Fernandez teaches away from the present invention. Specifically, the disclosed algorithm (Fernandez) provides for a method of flattening XML data into a relational tuple, whereas the present invention provides for a method of publishing an XML document from a relational tuple. Because Fernandez discloses a method for flattening XML data based on a matching, pre-defined XML view query templates, the simplicity of the XML view is exploited. If an XML view is similar to the structure of data stored in a relational database, the problem of rendering relational data into hierarchical XML data is trivial. The disclosed method (Fernandez) does not provide for arbitrarily joining XML user queries or resultant XML data.

The section referenced by the examiner in col. 12, lines 60 and 61 discloses a view tree representation of a view query. While Fernandez does disclose the use of a tree data structure, its usage is in the formulation of a relational database query, not the structuring of relational data. The disclosed view tree (Fernandez) is limited by pre-defined, existing view query templates and Skolem functions. The method of the present invention provides for arbitrary and dynamic views since a tagger tree graph is generated at run-time, and XML results are hierarchically tagged as they are produced.

The referenced sections therefore fails to provide the claimed tagger tree graph, tagger tree operator, and parse tree associated and make no mention of using a tagger graph to tag and structure the results of an SQL query to a relational database, and thus fails to provide a method

for evaluating parse trees associated with each tagger operator to tag of said XML language over said relational database in the manner of the present invention.

The examiner has recited the exact language of the remaining claims (3-4, 10-13, 18-21, 24-25, 31-34, and 39-43) and has suggested a correlation with the Fernandez reference. However, the examiner has not established the required *prima facia* case of obviousness. Specifically, the examiner has not set forth any elements in Fernandez that provide for the claimed features. It is our position that Fernandez does not describe or suggest these claimed features.

The claims of the present invention include at least the following elements not provided for, nor suggested, by Fernandez:

- tagger operators comprising any of a tagger input operator, a tagger scalar operator or a tagger aggregate operator
- tagger graph includes a tagger input operator for each level in a result XML tree of said XML query
- said tagger input operators execute in a sorted outer union mode
- tagger input operators comprise a shared tagger row stream
- tagger input operators execute in a node strip mode
- tagger operators comprises a tagger row stream
- each tagger operator performs any of a cr8_elem, a cr8_attr, a cr8_attr_list, a cr8_fragments or a cr8_fragment_list function
- each tagger operator implements a next method to produce a result row
- transforming said XML queries into a language-neutral intermediate representation;

- rewriting said language-neutral intermediate representation into an equivalent form easily translated into an SQL query;
- tagger graph is generated from said equivalent form.
- tagger graph includes a tagger input operator for each node in a result XML tree of said XML query.
- tagger input operators execute in a sorted outer union mode and said translating step produces a single SQL query to produce a single sorted outer union relational database result.
- A method of tagging results of an XML query over a relational database, as per claim 11, wherein a number of relational database tables of said relational database are mapped to a number of virtual XML documents and said XML queries are issued over said virtual XML documents.

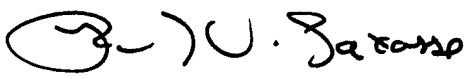
SUMMARY

As has been detailed above, none of the references, cited or applied, provide for the specific claimed details of applicant's presently claimed invention, nor renders them obvious. It is believed that this case is in condition for allowance and reconsideration thereof and early issuance is respectfully requested.

As this amendment has been timely filed within the set period of response, no petition for extension of time or associated fee is required. However, the Commissioner is hereby authorized to charge any deficiencies in the fees provided to Deposit Account No. 09-0441.

If it is felt that an interview would expedite prosecution of this application, please do not hesitate to contact applicant's representative at the below number.

Respectfully submitted,



Randy W. Lacasse
Registration No. 34,368

1725 Duke Street
Suite 650
Alexandria, Virginia 22314
(703) 838-7683
March 3, 2004